



# **Supply-Chain Risk Analysis**

**Bob Ellison, Chris Alberts,  
Rita Creel, Audrey Dorofee, and  
Carol Woody**



Report Documentation Page				Form Approved OMB No. 0704-0188	
Public reporting burden for the collection of information is estimated to average 1 hour per response, including the time for reviewing instructions, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington Headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington VA 22202-4302. Respondents should be aware that notwithstanding any other provision of law, no person shall be subject to a penalty for failing to comply with a collection of information if it does not display a currently valid OMB control number.					
1. REPORT DATE <b>09 JUN 2010</b>		2. REPORT TYPE		3. DATES COVERED <b>00-00-2010 to 00-00-2010</b>	
4. TITLE AND SUBTITLE <b>Supply-Chain Risk Analysis</b>				5a. CONTRACT NUMBER	
				5b. GRANT NUMBER	
				5c. PROGRAM ELEMENT NUMBER	
6. AUTHOR(S)				5d. PROJECT NUMBER	
				5e. TASK NUMBER	
				5f. WORK UNIT NUMBER	
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) <b>Carnegie Mellon University ,Software Engineering Institute,Pittsburgh,PA,15213</b>				8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)				10. SPONSOR/MONITOR'S ACRONYM(S)	
				11. SPONSOR/MONITOR'S REPORT NUMBER(S)	
12. DISTRIBUTION/AVAILABILITY STATEMENT <b>Approved for public release; distribution unlimited</b>					
13. SUPPLEMENTARY NOTES					
14. ABSTRACT					
15. SUBJECT TERMS					
16. SECURITY CLASSIFICATION OF:			17. LIMITATION OF ABSTRACT <b>Same as Report (SAR)</b>	18. NUMBER OF PAGES <b>51</b>	19a. NAME OF RESPONSIBLE PERSON
a. REPORT <b>unclassified</b>	b. ABSTRACT <b>unclassified</b>	c. THIS PAGE <b>unclassified</b>			

# Biography: Bob Ellison

---



Bob Ellison is a senior member of the technical staff of the CERT program at the Software Engineering Institute. He is currently the technical leader of a DHS funded project on supply-chain risks. He participated in the design and development of the DHS Build-Security-In Web site and continues to contribute articles to it. His recent work includes the development of the Survivability Analysis Framework which considers the affects of security threats on complex operational business processes. He is a coauthor of the book “Software Security Engineering: A Guide for Project Managers” (Addison-Wesley 2008)

# Polling Question #1

---

**How did you hear about this webinar?**

1. Social Media (i.e., LinkedIn, Twitter)
2. SEI Website
3. SEI Member Bulletin
4. Email invitation from the SEI
5. Website with webinar calendar (i.e., [www.webinar-directory.com](http://www.webinar-directory.com))

# Software Supply Chain

---

The network of stakeholders that contribute to the content of a software product or that have the opportunity to modify its content.

Comprehensive National Cybersecurity Initiative 11

# Polling Question #2

---

Has your organization had a problem with software malware in the last year?

Answers:

- Yes
- No
- Do not know

# What We Will Cover

---

Software supply-chain complexity: slides 6-8

Strategy: slides 10-18

Supply-chain risk example 20-40

Summary: slides 42-44

# Supply-Chain Risk Examples

---

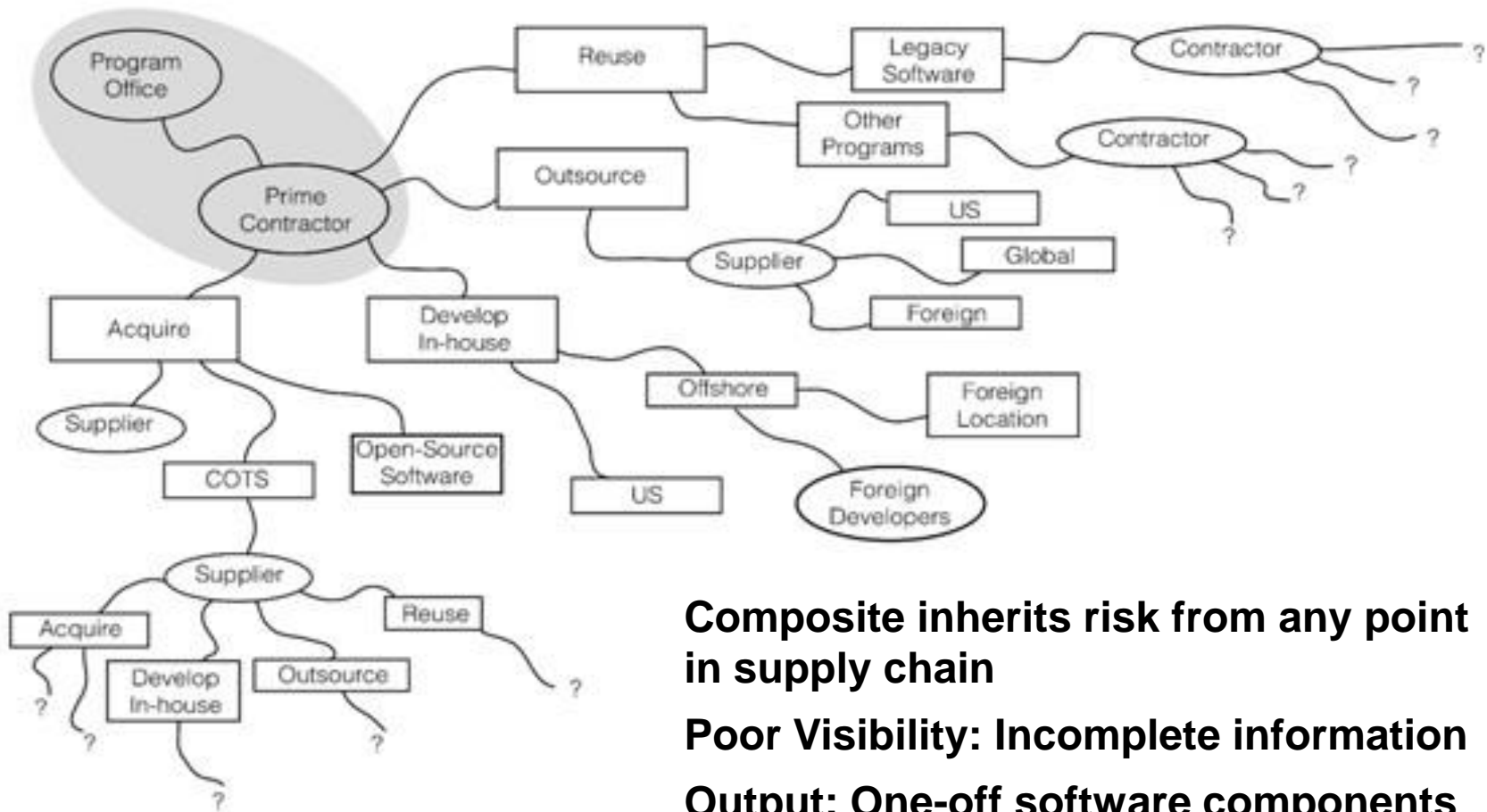
## Hardware

- Manufacturing and delivery disruptions
- Manufacturing quality
- Counterfeit hardware estimated at 10%
- Decades of data collection for physical supply chains

## Software

- Third-party tampering during development or delivery
- Malicious supplier
- Compromised by inadvertent introduction of exploitable design or coding errors
- Very little data for software supply chains

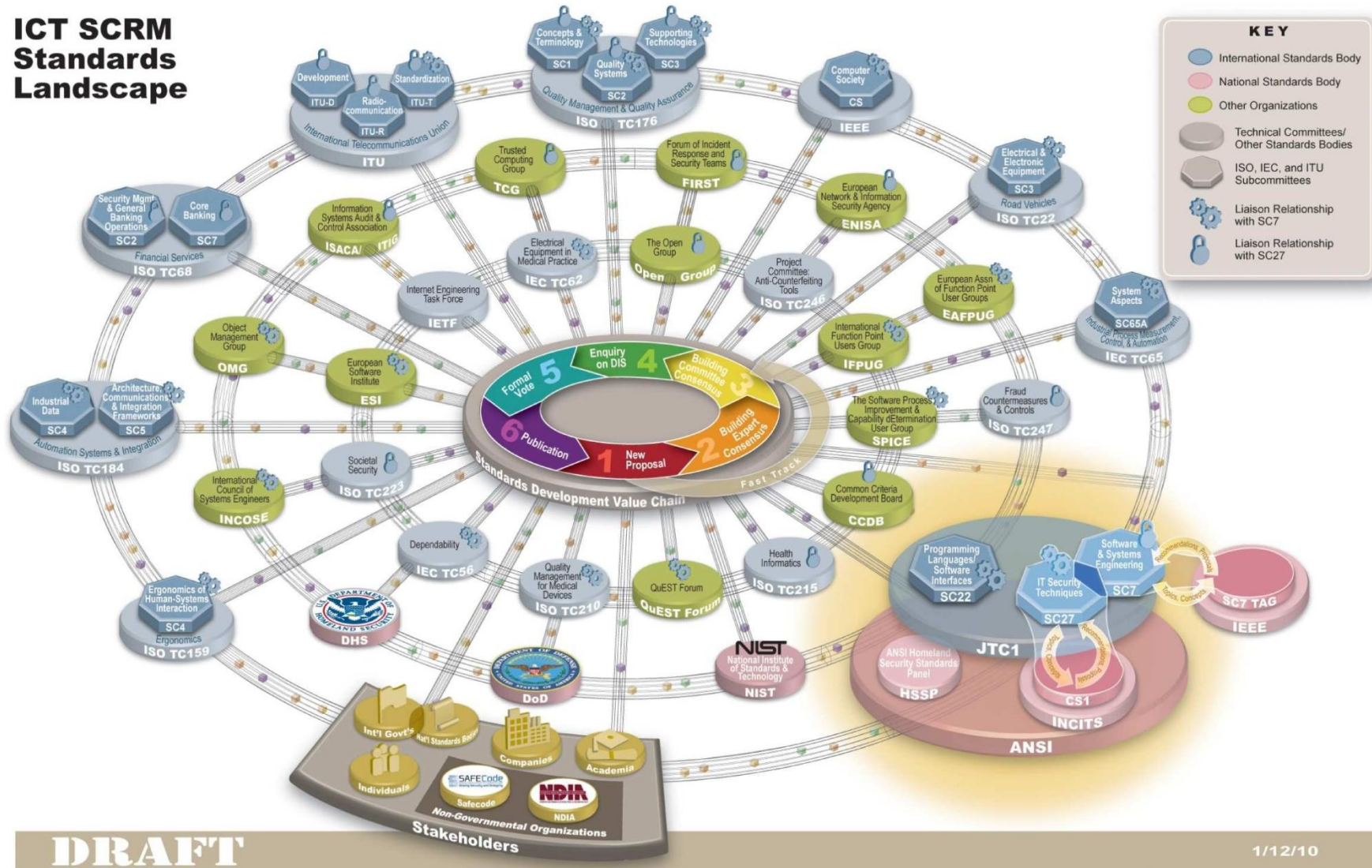
# Software Supply Chain Complexity<sup>1</sup>



# The Landscape

# Complexity<sup>3</sup>

## ICT SCRM Standards Landscape



Systems and Software Technology Conference 2010, Don Davidson, Globalization Task Force, DoD



# Strategy

# Propagation of Supply-Chain Risks

Selection	Evidence of Secure Software	Integration	Deployment Over time
<b>Construction</b> Secure Development Practices <b>Governance</b> Training Supplier and subcontractor management Verification of third-party software	Supplier and independent verifications Used recommended mitigations from CWE Weaknesses and mitigations tested Systematic testing of invalid input Static analysis of source code	Mitigation of risks not adequately addressed by supplier Effects of component supply-chain risk on aggregate system Risks induced by integration: Assumption mismatches Verify that aggregate risk is still acceptable	Install supplier updates Periodically update risk assessment: changes in usage, attack patterns, product updates, suppliers Monitor operational system behavior for unexpected events: test of design assumptions

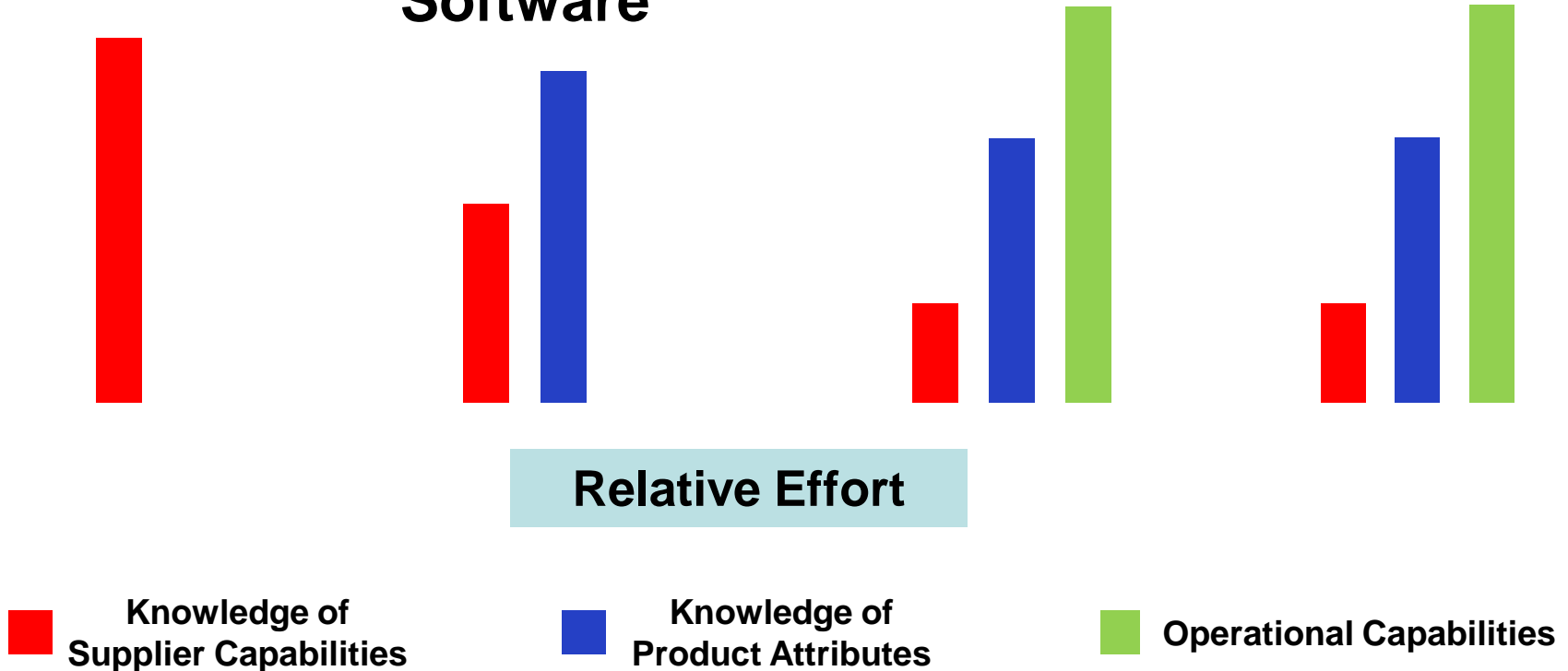
# Information Needs by Activity

## Evidence of Secure Software

Selection

Integration

Deployment



# Supply-Chain Risk Categories

---

Category	Description
Acquirer Capability	Operational preparedness, acquisition task execution, event management
Supplier Capability	Governance, Construction, Verification, Deployment
Product	An assessment of the problems and issues associated with a software product
Product Logistics	Access control of the software product at each step in the supply chain
Operational Product Control	Implementation of appropriate operational configuration and monitoring controls to reduce the risk of unauthorized changes to software products

# Strategy Outline<sup>-1</sup>

---

A solution depends on a combination of

- Supplier capabilities to create secure software
  - A necessity
- Product verification
  - What evidence shows that supplier expertise has been effectively applied to produce more secure software?
- Acquirer capabilities
  - Capability to manage multiple suppliers
  - Match software usage with supplier's intent
  - Manage changes in usage, suppliers, and attack patterns

# Strategy Outline<sup>-2</sup>

---

## Acquirer has to plan for security after deployment

- No guaranteed way to find maliciously inserted code
- Supply chain risk assessment can be invalidated by
  - New attack techniques and software weaknesses
  - Changes in acquirer usage that activate unused product features
  - Product upgrades that add features or change implementation
  - Increase in criticality with new or expanded usage
  - Changes in the supplier risk factors: mergers, corporate policies, staff training, development life cycle
- Operational management has to deal with incomplete supplier, product, and attack risk information

# Polling Question #3

---

Does your organization consider a vendor's capabilities to produce secure software when purchasing COTS software or outsourcing software development?

Answers:

- Yes
- No
- Do not know

# SEI Project

---

## Supply Chain Risk Model

- Develop a model that helps to structure and simplify analysis
- Initial focus on software supply chain
- Software supply chain risk management is more than a supplier assessment
  - Manage supply-chain risks that continue into deployment
  - Need increased understanding of allocation of responsibilities among suppliers and acquirers

# Supply Chain Drivers

---

A systemic risk assessment is based on a small set of factors that strongly influence the eventual outcome or result.

These factors are commonly referred to as drivers.

SEI experience shows that about 15-25 drivers are needed to establish a comprehensive profile of systemic risks to mission success.

These drivers reflect both supplier and acquirer factors.

# General Set of Supply-Chain Drivers

---

1. Software Supply-Chain Objectives
2. Acquisition Plan
3. Contracts
4. Development Process
5. Acquisition Task Execution
6. Coordination
7. Software Supply-Chain Interfaces
8. Information Management
9. Technology
10. Facilities and Equipment
11. Environmental Conditions
12. Compliance
13. Event Management
14. Requirements
15. Architecture
16. Design, Code, and Test
17. System Functionality
18. System Integration
19. Operational Support
20. Adoption Barriers
21. Operational Preparedness
22. System Risk Tolerance
23. Certification and Accreditation
24. Sustainment



# Software Supply-Chain Risk Example

# A Supply-Chain Weakness

---

Existing vulnerabilities present easy and effective opportunities for attackers – errors support malicious activities

Can reduce likelihood of vulnerabilities with incremental changes in development practices

- Draw from
  - Microsoft's Secure Development Life Cycle
  - SAFECode
  - Build Security In Maturity Model (BSIMM)
  - Build-Security-In <https://buildsecurityin.us-cert.gov/daisy/bsi/home.html>

# Prevalence of Software Errors

---

MITRE has documented software errors that have led to exploitable vulnerabilities: Common Weakness Enumeration (CWE)

*CWE/SANS<sup>1</sup> Top 25 Most Dangerous Programming Errors* published yearly by MITRE – 3/1/2010

## Examples

Improper Input Validation

Cross-site scripting

Download of Code Without Integrity Check

Race Condition

SQL Injection

Use of Hard-coded Credentials

Improper Check for Unusual or Exceptional Conditions

Classic Buffer Overflow

1. <http://cwe.mitre.org/top25/>

SANS (SysAdmin, Audit, Network, Security) Institute

# Veracode: State of Software Security

---

58% of all applications did not achieve an acceptable security score upon first submission – 3/1/2010

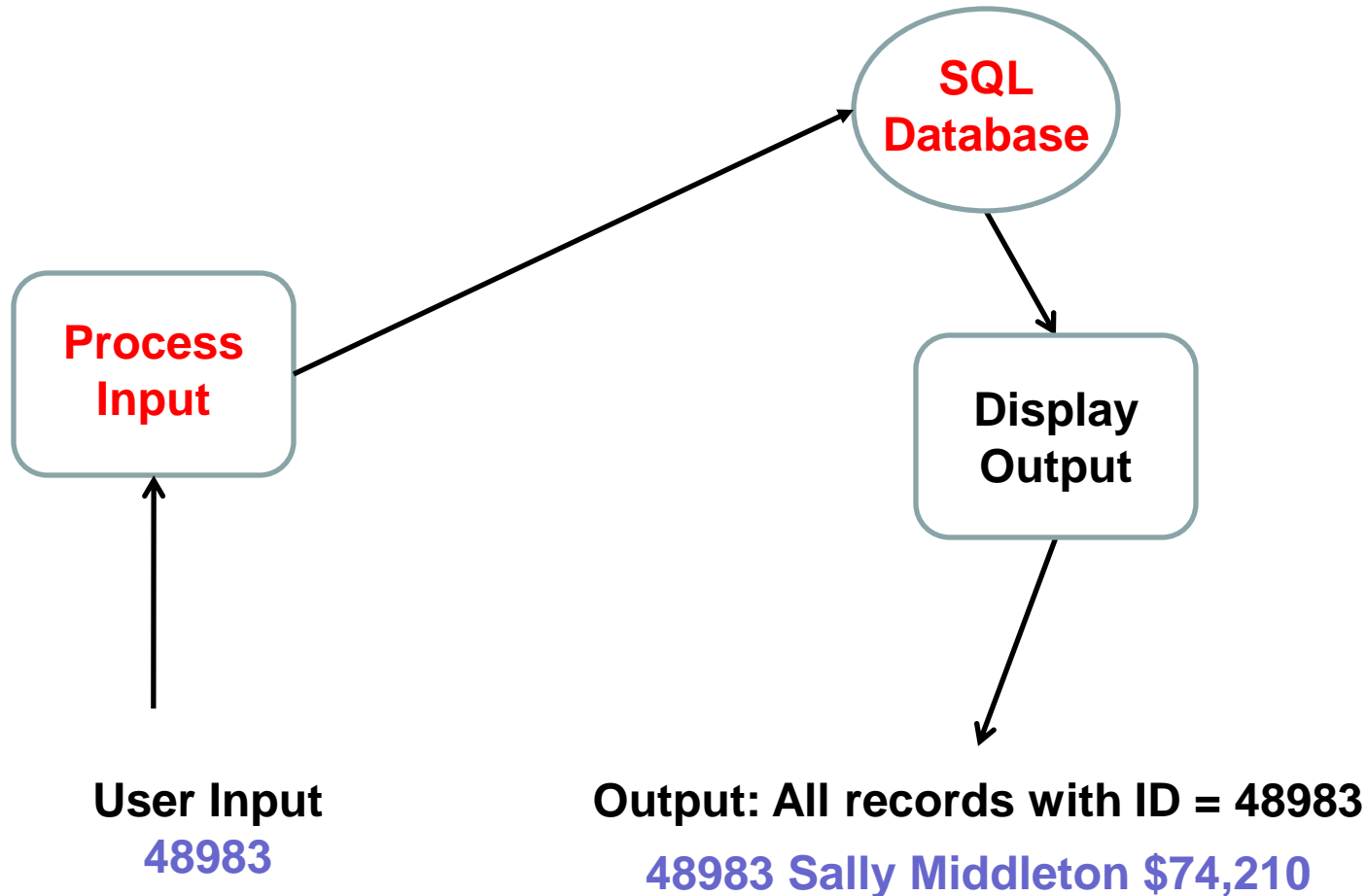
## Measured Against CWE/SANS Top-25 Errors

Software Source	Acceptable
Outsourced	6%
Open Source	39%
Internally Developed	30%
Commercial	38%

Veracode: The pervasiveness of easily remedied weaknesses suggests developer training for secure software development is a critical supplier criteria.

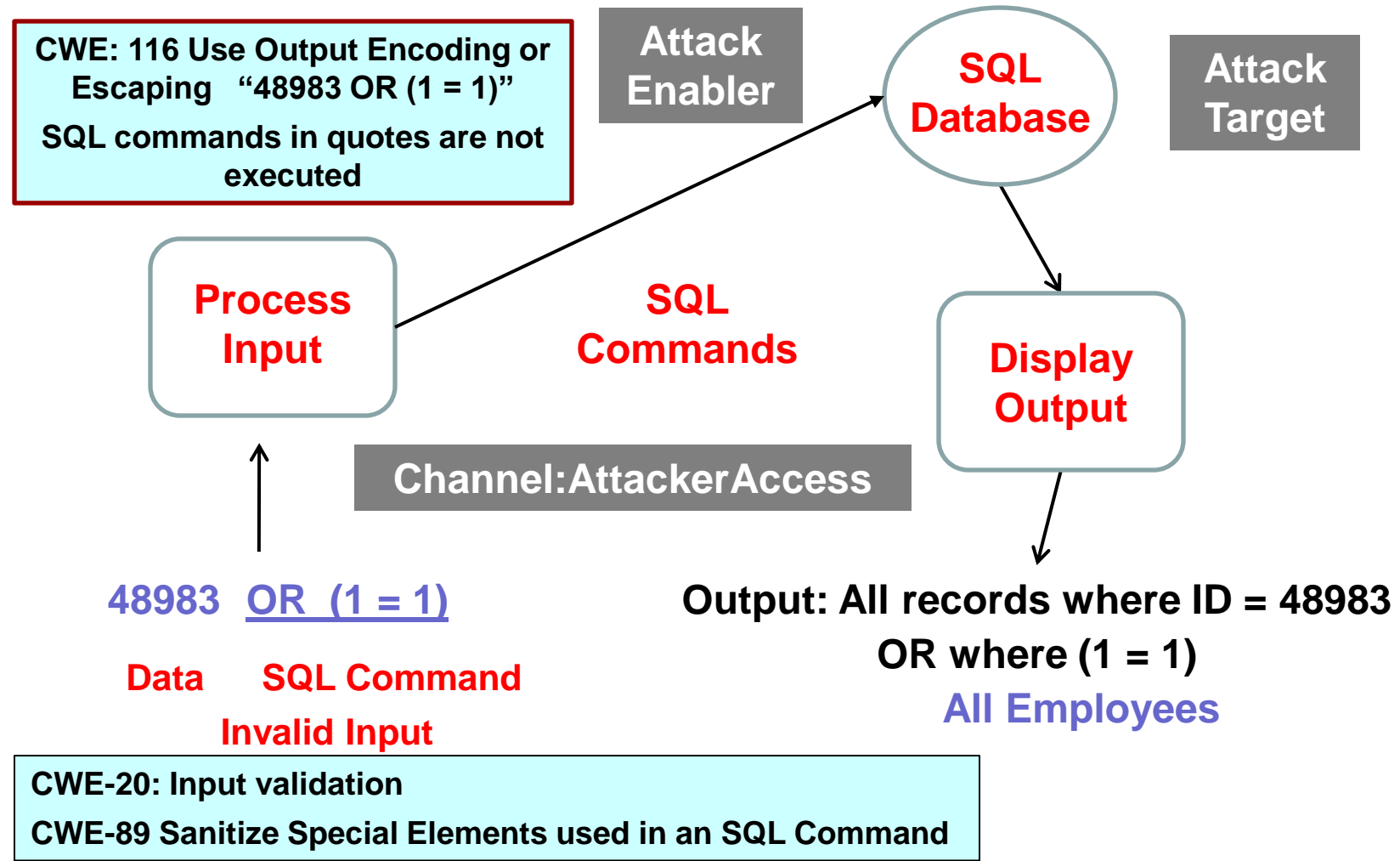
# SQL Database Query

---



Could involve multiple supply chains: web server, SQL database, and contracted software development

# CWE-89: Attacker View - SQL Injection



# Assessments By Activity

---

## Selection

Relative  
Effort



## Construction

Secure Development Practices

## Governance

Training

Supplier and subcontractor  
management

Verification of third-party software



Knowledge of  
Supplier Capabilities



Knowledge of  
Product Attributes

# Driver: Design, Code and Test

---

***Is the code's quality sufficient to meet system requirements and provide the desired operational capability***

---

Design reviews

Source code reviews

Coding practices

Static code analysis

Unit and integration testing

Analysis of common  
weaknesses

Analysis of attack patterns

Threat/vulnerability analysis

Software security testing

Dynamic testing

Code interfaces and dependencies

---

# Evidence of Secure Software

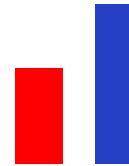
---

**Selection**

**Verification**



**Relative  
Effort**



## **Evidence of Secure Software**

Supplier and/or independent verifications

Used recommended mitigations

Likely software weaknesses and  
mitigations tested

Systematic testing of invalid input

Static analysis of source code



**Knowledge of  
Supplier Capabilities**



**Knowledge of  
Product Attributes**

# Product Evidence: Testing

---

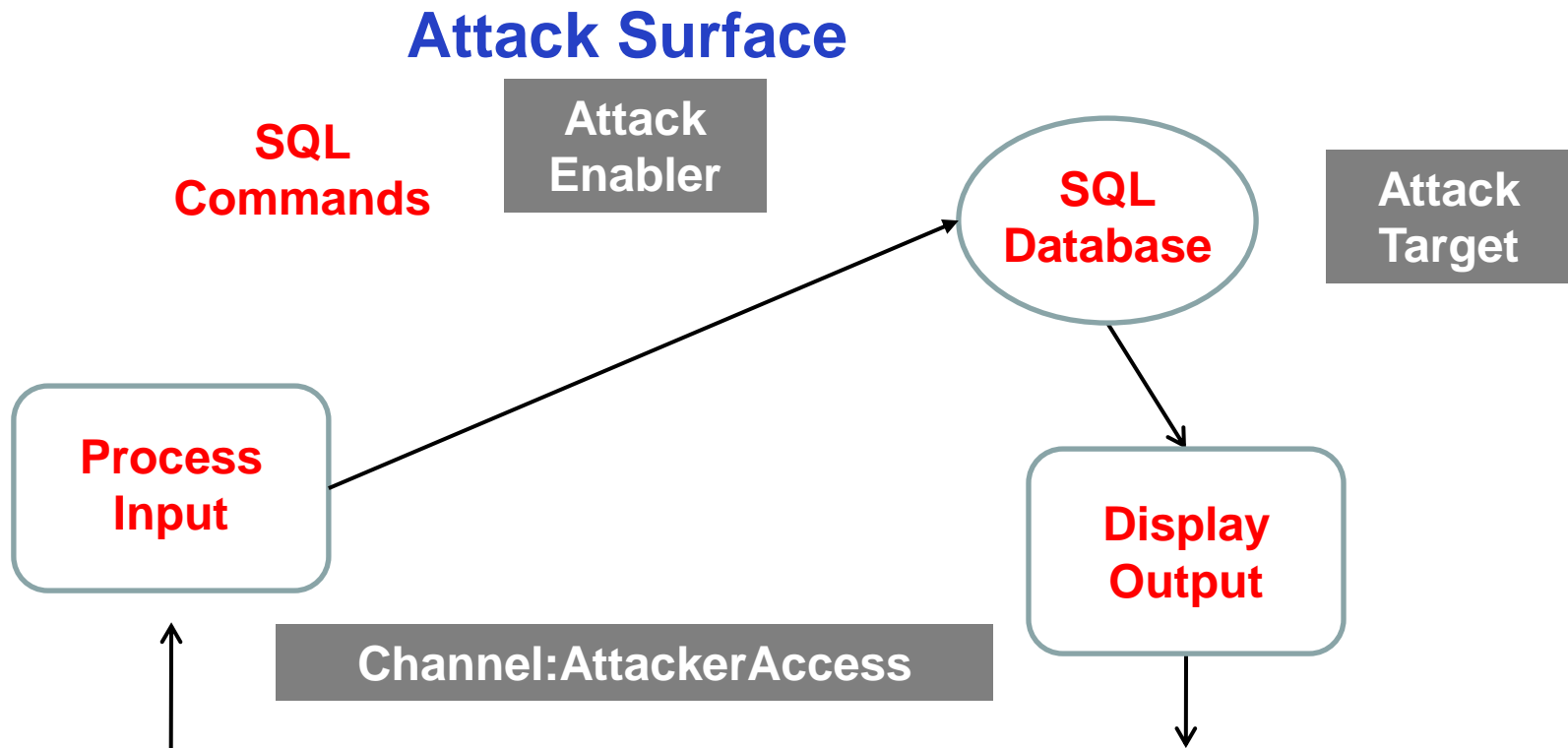
## Security Testing

- Potential software weaknesses and mitigations tested
- Systematic testing of invalid input – fuzz testing
- Static analysis of source code

## Testing is increasingly automated and outsourced

- Limited value for risk analysis:
  - We know neither the consequences or likelihood for any remaining vulnerabilities nor the costs and effectiveness of possible mitigations
- Expensive redesign and mitigations: Veracode statistics on initial failures for security testing.

# Product Evidence: Attackability



A system with more targets, more enablers, more channels or more generous access rights provides more opportunities to the attacker.

**Attack surface:** targets, enablers( exploitable features), communication channels, and access controls

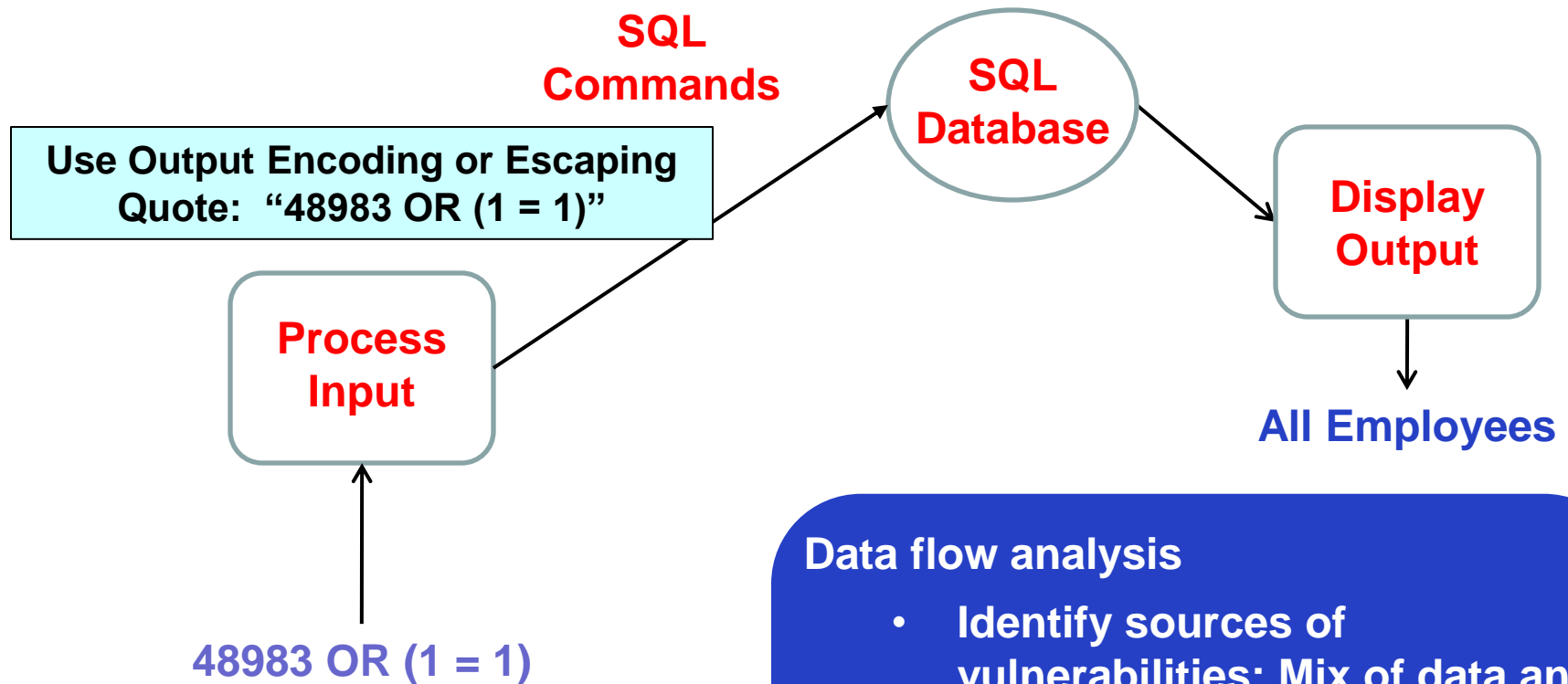
# Using Attack Surface Analysis

---

## Reduce Attack Surface

- Remove or change system features or re-architect the implementation to avoid attack enablers or unnecessary channels.
- Revise use of an emerging technology where there is limited knowledge of the potential exploits and mitigations
- Review requirements or implementation if existing mitigations are costly or do not provide the necessary assurance

# Data Flow Analysis



## Data flow analysis

- Identify sources of vulnerabilities: Mix of data and commands
- Consider consequences
- Analyze mitigations
- Provide architecture and design guidance

# Data Flow Analysis Benefits

---

## Supports

- Objective trade-off discussions involving security risks during initial development or with later upgrades
- Supply-chain risk management – consequences and mitigations
- Traceability and business justifications
- System integration – insight into design assumptions, attack patterns considered and mitigation strategy
- Operational monitoring – design assumptions about expected behavior

# Threat Modeling

---

Threat Modeling: During a data flow walk through

- Document security assumptions and trust boundaries
- Consider known weaknesses and attack patterns
- Consider deployed configuration and expected usage
- Analyze the interfaces to other components (inputs and outputs)
- Analyze possible mitigations

Value recognized – Microsoft's SDL, BSIMM  
collection of current practices drawn from thirty  
firms

See Stevens (references) for adoption considerations

# Driver: Acquisition Task Execution

---

***Are tasks and activities performed effectively and efficiently?***

---

Experience and  
expertise of  
management and staff

Sufficient experience in software  
security, reliability, and safety  
engineering

Resources allocated to  
tasks and activities

Experience with software supply  
chains

---

# Polling Question #4

---

Do your suppliers and in-house developers incorporate threat modeling as part of the vulnerability analysis?

Answers:

- Yes
- No
- Do not know

# Incorporate into Acquisition: RFP

---

## RFP: ask for evidence

- Development staff training
- Documentation of potential attacks and mitigations
- Supplier capabilities as demonstrated with development of other systems
- For contracted development, require application of threat modeling to analyze risks associated with architecture and design decisions

# Driver: Contracts

---

***Are the contract mechanisms with each participating group or team sufficient?***

**Includes suppliers contracts with their suppliers or subcontractors**

---

Acquisition and  
development strategies

Sufficient focus on software  
security, reliability, and safety

Resources

Contracts with each participating  
group or team

Funding

Schedule

Intellectual property  
considerations

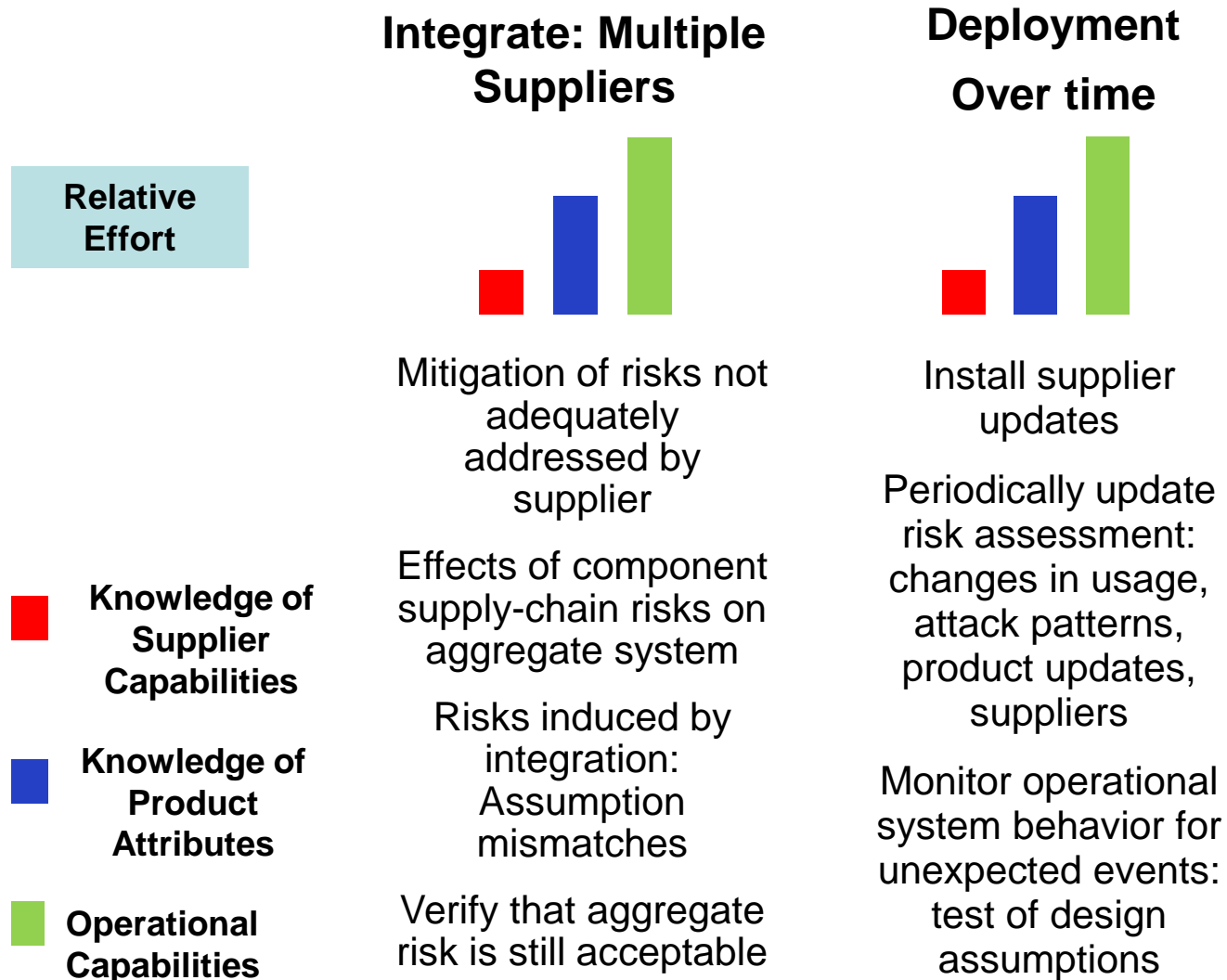
Alignment among the contracts  
of participating groups or teams

Licensing agreements

Roles and responsibilities

---

# Integration and Deployment



# Driver: System Integration

---

***Will the system sufficiently integrate and interoperate with other systems when deployed?***

---

Interfaces	COTS software
Applications	Performance, security, reliability, and safety of the integrated system
Tools	Failure analysis
Hardware	Security testing
Data	Legacy systems

---

# Driver: Event Management

---

***Does the software supply chain have sufficient capacity and capability to identify and manage potential events and changing circumstances?***

---

Expected and unexpected potential events and changing circumstances

Program continuity, disaster, and contingency plans

Changes in personnel or suppliers

Issue/problem management plan, process, and tools

Changes in product usage

Changes in requirements

---



# Summary

# Manage Supply-Chain Risk

**Operational Context**, e.g., usage, requirements, operational preparedness, risk tolerance

**Acquisition Scope**, e.g., product, system, system of systems, major upgrade, component replacement

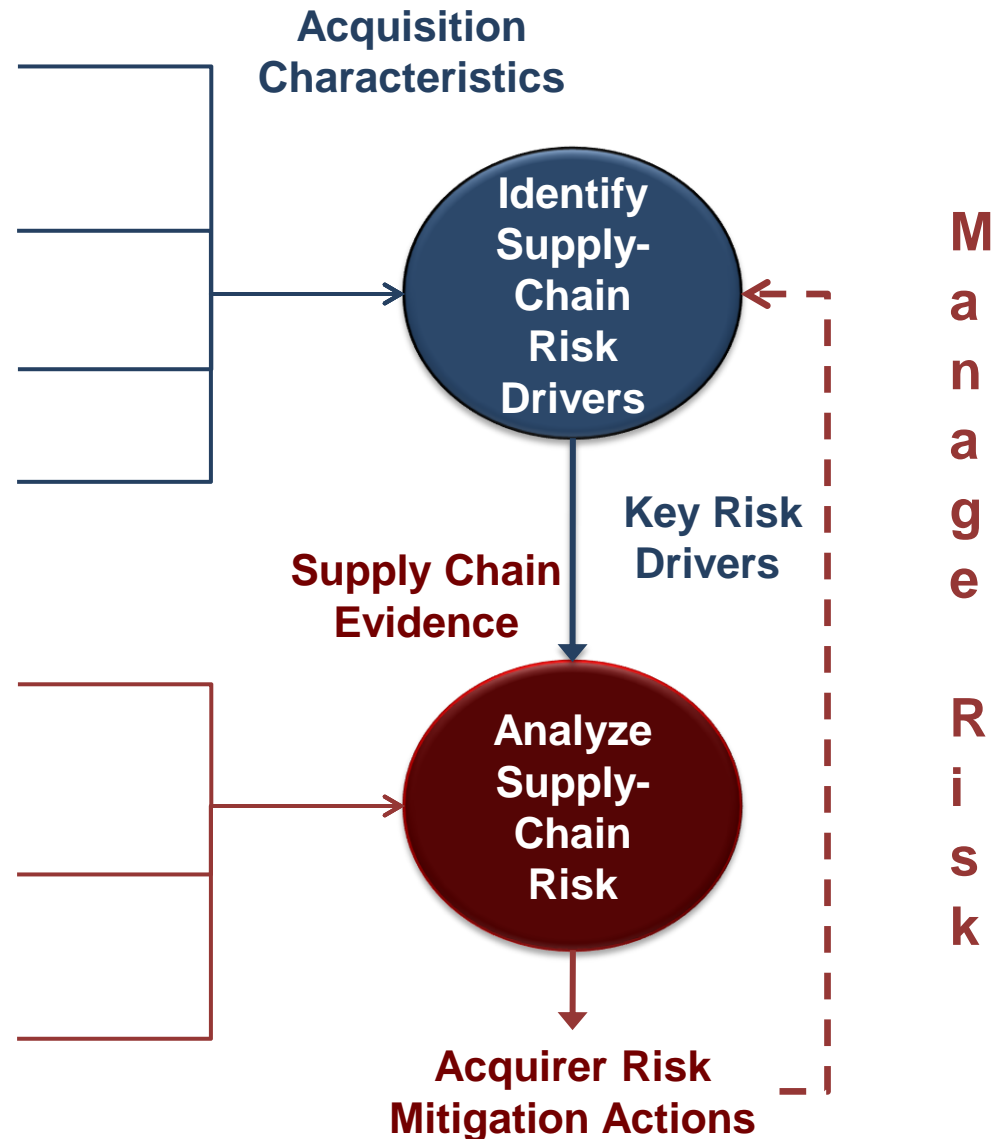
**Supplier Capability Data**, i.e., guidance for supplier evaluation

**Preliminary Product Data**, i.e., guidance for product evaluation

**Supplier Product Development Information**, e.g., architecture, design-code-test, compliance, supply-chain interfaces, event management

**Acquirer Information**, e.g., acquisition plan, acquisition task execution, event management

**Operational Product Control**, i.e., monitoring and configuration control of software products



# Summary

---

Supplier, acquirer, and operator all have roles to ensure good practices are applied!

A supply-chain risk model helps to manage complexity and provides a structure for risk analysis

Example: Remove widely exploited software weaknesses with known mitigations

- Feasible
- Incremental changes to existing software development and acquisition life cycles
- Demonstrated value

# Sources

---

## Evaluating and Mitigating Software Supply Chain Security Risks

- <http://www.sei.cmu.edu/library/abstracts/reports/10tn016.cfm>

## Attack Surface

- Michael Howard, 2003, <http://msdn.microsoft.com/en-us/library/ms972812.aspx>

## Threat Modeling

- Frank Swiderski, Window Snyder, *Threat Modeling*, 2004
- Michael Howard and Steve Lipner. *The Security Development Lifecycle*, 2006
- James McGovern, & Gunnar Peterson. “10 Quick, Dirty, and Cheap Things to Improve Enterprise Security.” *Security & Privacy*, IEEE, March-April 2010
- Building Security In Maturity Model (BSIMM) <http://bsimm2.com/index.php>
- John Stevens, “Threat Modeling— Perhaps It’s Time”, *Security & Privacy*, IEEE, May-June 2010

---

## NO WARRANTY

THIS MATERIAL OF CARNEGIE MELLON UNIVERSITY AND ITS SOFTWARE ENGINEERING INSTITUTE IS FURNISHED ON AN "AS-IS" BASIS. CARNEGIE MELLON UNIVERSITY MAKES NO WARRANTIES OF ANY KIND, EITHER EXPRESSED OR IMPLIED, AS TO ANY MATTER INCLUDING, BUT NOT LIMITED TO, WARRANTY OF FITNESS FOR PURPOSE OR MERCHANTABILITY, EXCLUSIVITY, OR RESULTS OBTAINED FROM USE OF THE MATERIAL. CARNEGIE MELLON UNIVERSITY DOES NOT MAKE ANY WARRANTY OF ANY KIND WITH RESPECT TO FREEDOM FROM PATENT, TRADEMARK, OR COPYRIGHT INFRINGEMENT.

Use of any trademarks in this presentation is not intended in any way to infringe on the rights of the trademark holder.

This Presentation may be reproduced in its entirety, without modification, and freely distributed in written or electronic form without requesting formal permission. Permission is required for any other use. Requests for permission should be directed to the Software Engineering Institute at [permission@sei.cmu.edu](mailto:permission@sei.cmu.edu).

This work was created in the performance of Federal Government Contract Number FA8721-05-C-0003 with Carnegie Mellon University for the operation of the Software Engineering Institute, a federally funded research and development center. The Government of the United States has a royalty-free government-purpose license to use, duplicate, or disclose the work, in whole or in part and in any manner, and to have or permit others to do so, for government purposes pursuant to the copyright license under the clause at 252.227-7013.



**SEPG<sup>SM</sup> 2010**  
**NORTH AMERICA**

*Perform at a Higher Level*

# SEPG Europe 2010

28 June - 1 July, 2010

Porto, Portugal



[www.sei.cmu.edu/sepg/europe/2010/](http://www.sei.cmu.edu/sepg/europe/2010/)

**SEPG** is the premier, global conference series on software and systems process management



Want a Closer Connection to the SEI?

Become an SEI Member!

▶ [www.sei.cmu.edu/membership](http://www.sei.cmu.edu/membership)



Do you have the knowledge you need?

SEI Training

▶ [www.sei.cmu.edu/training](http://www.sei.cmu.edu/training)

	Name	Time	Art
1	Convergence: Integrating Physica...	28:43	B, C
2	IT Infrastructure: Tips for Navigat...		
3	The Value of De-Identified Perso...		
4	Adapting to Changing Risk Enviro...		

**CERT's Podcast Series: Security for Business Leaders**

[www.cert.org/podcast/](http://www.cert.org/podcast/)

A world map with a blue-to-green color gradient, showing the continents and oceans. The map is centered on the Atlantic Ocean, with North and South America on the left and Europe, Africa, and Asia on the right.

For more than 20 years, the SEI has been at the forefront of software engineering.

By becoming an SEI Partner, you join forces with a software engineering pioneer and an institute whose credibility provides a solid foundation during uncertain economic times.

SEI Partner Network

▶ [www.sei.cmu.edu/partners](http://www.sei.cmu.edu/partners)